

REMARKS

By this amendment, no claims have been added, cancelled, or amended. Hence, Claims 8-30, 32-35, 37-40, and 42-44 are pending in the application.

While each issue raised by the Office Action is addressed below, Applicants initially wish to bring to the Examiner's attention that Claim 20 has been rejected without any arguments being presented on the record as to why certain elements recited by Claim 20 are allegedly shown by the Office Action. Consequently, if the present application is not allowed in the next action, Applicants respectfully submit that the finality of the present Office Action be removed.

CLAIMS 8-30, 32-35, 37-40, AND 42-44

Claims 8-30, 32-35, 37-40, and 42-44 were rejected under 35 U.S.C. § 103(a) as being anticipated by U.S. Patent No. 6,253,236 issued to Troxel et al. ("*Troxel*") in view of U.S. Patent No. 6,493,804 issued to Soltis et al. ("*Soltis*") in further view of U.S. Patent No. 5,913,213 issued to Wikstrom ("*Wikstrom*"). The rejections are traversed.

CLAIM 8

Claim 8 features:

A method of controlling use by concurrent users of a distributed resource on a network, wherein use of the resource is limited to a specified maximum number of concurrent users, the method comprising the computer-implemented steps of:
providing a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts,
wherein each of the plurality of local lock manager processes may grant a lock on the same resource;
associating a user identification for each user with one host of the plurality of hosts; and
responding to a request for the resource associated with a first user having a first user identification associated with a first host of the plurality of

hosts by requesting a lock from a first local lock manager process executing on the first host. (emphasis added).

Even if *Troxel*, *Soltis*, and *Wikstrom* were to be properly combined, at least the above-bolded elements of Claim 8 would still not be disclosed, taught, or suggested by *Troxel* or *Soltis* or *Wikstrom*, either individually or in combination.

Claim 8 recites controlling use by concurrent users of a distributed resource on a network. The resource is limited to a specified maximum number of concurrent users. A distributed lock manager process, comprising a plurality of local lock manager processes, executes on a corresponding plurality of hosts. Each of the plurality of local lock manager processes may grant a lock on the same resource. A user identification for each user is associated with one host of the plurality of hosts. A request, associated with a first user having a first user identification associated with a first host of the plurality of hosts, for the resource is responded to by requesting a lock from a first local lock manager process executing on the first host. By responding to requests for resources in this fashion, the number of concurrent users of a distributed resource on a network may be controlled in a manner that scales to support a large number of clients and avoids a single point of failure.

Troxel, on the other hand, describes enabling a client computer system to access data on a mainframe host computer system in a manner that prevents conflicts for data stored under different types of file access methods and provides security at a file level and recovery from interruption of communication between the host and client computer systems (Col. 2, line 40 – Col. 3, line 21). *Troxel* maintains a concurrent user counter that is incremented each time a client computer system logs into the host computer system to enforce limits on the number of concurrent users that can access the host computer system (See concurrent user counter 354 at Col. 5, lines 26-30).

Troxel lacks any suggestion of a **distributed lock manager process** comprising a **plurality of local lock manager processes**. Instead, the cited portion of *Troxel* teaches a single, centralized component, on the mainframe host computer system, that is responsible for locking resources upon receipt of a client request (Col. 1, lines 15-36). Further, *Troxel* lacks any suggestion of associating a user identification for each user with one host of a plurality of hosts. Also, the cited portion of *Troxel* lacks any suggestion of responding to requests for resources as claimed. In recognition of the deficiencies of *Troxel*'s teachings, the Office Action acknowledges (see page 3):

Troxel does not explicitly teach providing a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts; associating a user identification for each user with one host of the plurality of hosts; and responding to a request for the resource associated with a first user having a first user identification associated with a first host of the plurality of hosts by requesting a lock from a first local lock manager process on the first host.

Thus, the Office Action acknowledges that *Troxel* does not teach or suggest ANY of the above-bolded elements of Claim 8.

The Office Action also cites *Soltis* in rejecting Claim 8. *Soltis* teaches an approach for a global file system that comprises a plurality of clients which each may access one or more data storage devices of the global file system (See FIG. 1; Col. 5, lines 25-45; Col. 10, lines 9-11). The data storage devices may be pooled together into a shared disk memory in a Network Storage Pool (NSP) arrangement (Col. 10, lines 19-23). A data storage device includes one or more locks that are each associated with the use of a storage block of the data storage device. Clients access a particular storage block, of a particular data storage device, by requesting a lock

associated with the particular storage block from the particular storage device (see Abstract; Col. 2, line 45 - Col. 3, line 64).

In the approach of *Soltis*, the Network Storage Pool (NSP) provides shared storage resources that are substantially equally accessible to each client in the system (Col. 10, lines 33-35). *Soltis* teaches that each data storage device maintains a set of locks that may be granted to any of the clients accessing data blocks of only that data storage device (see FIG. 7; Col. 13, line 65 – Col. Col. line 32). **A particular data storage device of *Soltis* may only grant locks on the data block of that particular data storage device.** Thus, *Soltis* expressly teaches away from a distributed lock manager process comprising a plurality of local lock manger processes, executing on a corresponding plurality of hosts, that each may grant a lock on the same resource.

As a result, *Soltis* cannot teach a plurality of local lock manager processes that may each grant a lock on the same resource.

Perhaps in acknowledgement of this deficiency in *Soltis*, the Office Action cites a third reference, namely *Wikstrom*, to show “wherein each of the plurality of local lock manager processes may grant a lock on the same resource.” However, *Wikstrom* also fails to teach or suggest this feature. Instead, *Wikstrom* describes only a single lock manager residing at a node, and each lock manager may only grant locks on resources that reside on the same node as the lock manager.

Wikstrom describes locking **replicated** data objects (see title). The fact that *Wikstrom* describes locking replicated data objects **teaches away** from the approach of Claim 8. In *Wikstrom*, each node of a plurality of nodes has its own lock manager. For example, FIG. 1 of

Wikstrom shows node 30A comprises lock manager 73A and node 30B comprises lock manger 73B. In *Wikstrom*, each node has its own version of a replicated data object. For example, *Wikstrom* states:

Each node 30 has its own version of a data object X. Specifically, node 30A has hard disk 32A whereon its version of data object X, referenced as X-A, is stored. Similarly, node 30B has hard disk 32B whereon its version of data object X, referenced as X-B, is stored (see Col. 3, lines 38-43).

Further, in *Wikstrom*, each lock manager operates on the version of a replicated data object that resides at the same node as the lock manager. For example, *Wikstrom* states:

Since versions of data object X are stored both at nodes 30A and 30B, when one node updates the value of data object X, the updated value is communicated to the other node so that the other node can likewise have the updated value, thereby maintaining a coordination of the value of the data object X. (see Col. 3, lines 47-53).

As illustrated in FIG. 2, object lock table 220 has a record 220 for each data object pertinent to node A. (see Col. 4, lines 42-43. This teaching shows that the lock table for node A only maintains records for resources at node A).

Consequently, in *Wikstrom*, a single lock manager resides at a node, and each lock manager may only grant locks on resources that reside on the same node as the lock manager. As a result, *Wikstrom* cannot possible teach or suggest the feature of **“wherein each of the plurality of local lock manager processes may grant a lock on the same resource.”**

The Office Action argues that *Wikstrom* teaches this feature at Col. 6, lines 38-44; lines 54-64, and FIG. 5. However, this portion of *Wikstrom* merely describes sending a lock request message to each lock manager (LM) residing at each node. When each lock manager receives a lock request message, each lock manager locks a different resource, namely a particular version of a data object that resides at the same node as the lock manager. Nothing in the cited portion of *Wikstrom* teaches or suggests that two or more lock managers may grant locks on the same

resource. If the Office disagrees, the Office is respectfully requested to specifically identify, within the teachings of *Wikstrom*, (a) the resource (and the location of the resource) that is the subject of the lock granted by multiple local lock managers, and (b) the two or more local lock managers that may each grant a lock on the same resource.

Consequently, *Wikstrom* also cannot teach a plurality of local lock manager processes that may each grant a lock on the same resource.

As a result of the fundamental differences between Claim 8 and the approach of *Troxel*, *Soltis*, and *Wikstrom*, numerous features of Claim 8 are not disclosed, taught, or suggested by *Troxel*, *Soltis*, or *Wikstrom*, either individually or in combination.

For example, Claim 8 recites the feature of “*providing a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts*” and “*wherein each of the plurality of local lock manager processes may grant a lock on the same resource.*” The single, centralized server of *Troxel* that is responsible for locking resources upon receipt of a client request (Col. 1, lines 15-36) does not suggest a plurality of local lock manger processes, executing on a corresponding plurality of hosts, that each may grant a lock on the same resource. Indeed, the Office Action acknowledges, “*Troxel* does not explicitly teach providing a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts,” and instead, relies upon *Soltis* to show these features.

The portion of *Soltis* cited to show these features (FIG. 4; Clients 105A-N) merely shows a plurality of clients. Clients 105A-N of FIG. 4 request locks on data blocks from the data

storage devices in the Network Storage Pool (NSP) 400. While each of the data storage devices, in the NSP 400, may grant a lock on a data block, a particular data storage device only grants locks to data blocks of that particular data storage device (see FIG. 7; Col. 13, line 65 – Col. Col. line 32). Thus, *Soltis* expressly teaches away from the feature of “wherein each of the plurality of local lock manager processes may grant a lock on the same resource.”

Moreover, *Wikstrom* cannot teach or suggest this feature as well, because in *Wikstrom*, each lock manager receives a lock request message, each lock manager locks a different resource, namely a particular version of a data object that resides at the same node as the lock manager. Thus, the above-quoted elements are not disclosed, taught, or suggested by *Troxel*, *Soltis*, or *Wikstrom*, either individually or in combination.

Claim 8 also recites the feature of “*associating a user identification for each user with one host of the plurality of hosts.*” The portion of *Soltis* cited to show this feature (FIG. 4; Clients 105A-N; FIGS. 9 and 10, Abstract, Col. 2, lines 47 to Col. 3, line 20) merely shows a plurality of clients. No explanation is provided by the Office Action as to why clients 105 A-N of FIG. 4 show associating a user identification for each user with one host of the plurality of hosts. Importantly, the entire disclosure of *Soltis* lacks any teaching of associating a user identification for each user with any host. Further, each host, of the plurality of hosts, to which the user identifications are associated, is executing a local lock manager process. As explained above, *Soltis* lacks any teaching of a local lock manger process as claimed; consequently, it follows that *Soltis* lacks any teaching of associating a user identification for each user with one host of the plurality of hosts that are each executing a local lock manager process as claimed. Thus, this element cannot be disclosed, taught, or suggested by *Soltis*. The Office Action

acknowledges that *Troxel* fails to teach this element, and does not cite *Wikstrom* to show this element.

Claim 8 also recites the feature of “*responding to a request for the resource associated with a first user having a first user identification associated with a first host of the plurality of hosts by requesting a lock from a first local lock manager process executing on the first host.*” *Soltis* is cited to show this element. However, as explained above, *Soltis* lacks any teaching or suggestion of (a) a local lock manager as claimed, (b) a user identification as claimed. Thus, *Soltis* cannot possibly teach or suggest this element, which involves a local lock manager and a user identification.

At best, the portion of *Soltis* cited to show this feature discusses that clients may acquire locks for shared use by multiple clients. However, *Soltis* teaches that a particular data storage device may only grant locks for data blocks of the particular data storage device (see FIG. 7; Col. 13, line 65 – Col. Col. line 32). Thus, when a data storage device receives a request for a resource that involves a lock, rather than requesting a lock from a local lock manager process executing on a host associated with the user identification that is associated with the requesting user, the data storage device processes the request without communicating with a local lock manager process. As a result, this element is not disclosed, taught, or suggested by *Soltis*. The Office Action acknowledges that *Troxel* fails to teach this element, and does not cite *Wikstrom* to show this element.

As at least one element of Claim 8 is not disclosed, taught, or suggested by *Troxel*, *Soltis*, or *Wikstrom*, either individually or in combination, it is respectfully submitted that Claim 8 is patentable over the cited art and is in condition for allowance.

Moreover, *Troxel*, *Soltis*, and *Wikstrom* also have not been properly combined, and as a result, the rejection under 35 U.S.C. § 103(a) may not be properly maintained. The Office Action states that it would have been obvious to “combine the teachings of *Troxel* with the teaching of *Soltis* and *Wikstrom* in order to provide decentralized control of shared data, therefore maintaining data consistency¹,” but nothing in either *Troxel*, *Soltis*, or *Wikstrom* teaches or suggests combining their respective teachings².

As stated in the Federal Circuit decision *In re Dembiczak*, 50 USPQ.2d 1617 (Fed. Cir. 1999), (citing *Gore v. Garlock*, 220 USPQ 303, 313 (Fed. Cir. 1983)), “it is very easy to fall victim to the insidious effect of the hindsight syndrome where that which only the inventor taught is used against its teacher.” *Id.* The Federal Circuit stated in *Dembiczak* “that the best defense against subtle but powerful attraction of a hindsight-based obviousness analysis is rigorous application of the requirement for a showing of the teaching or suggestion to combine prior art references.” *Id.* Thus, the Federal Circuit explains that a proper obviousness analysis requires “***particular factual findings*** regarding the locus of the suggestion, teaching, or motivation to combine prior art references.” *Id.* (emphasis added).

In particular, the Federal Circuit states:

“We have noted that evidence of a suggestion, teaching, or motivation to combine may flow from the prior art references themselves, the knowledge of one of ordinary skill in the art, or, in some cases, from the nature of the problem to be solved...although ‘the suggestion more often comes from the teachings of the pertinent references’...The range of sources available, however, does ***not diminish the requirement for actual evidence***. That is, the ***showing must be clear and particular***...Broad conclusory statements

¹ This is exactly the same motivation supplied previously, word for word, except now the motivation includes a third reference. This type of boilerplate motivation to combine is a textbook example of an impermissible hindsight-based rejection.

² The Office Action also completely fails to show how the approaches of *Troxel*, *Soltis*, and *Wikstrom* may possibly be used in conjunction with one another. For example, a single entity performs locking functions in *Troxel* and multiple entities perform locking functions in *Soltis* and *Wikstrom*. As another example of how it is unclear how the references may be combined, the resources being locked are different, e.g., in *Troxel*, files are locked, in *Soltis*, blocks are locked, and in *Wikstrom*, replicated data objects are locked.

regarding the teaching of multiple references, standing alone, are not ‘evidence.’” *Id.* (emphasis added; internal citations omitted).

Neither *Troxel*, nor *Soltis*, nor *Wikstrom* show any suggestion, teaching, or motivation to combine their teachings, nor does the Office Action provide a “clear and particular” showing of the suggestion, teaching, or motivation to combine their teachings. Nor portion of any reference is cited to support a motivation to combine. In fact, the only motivation provided in the Office Action is the hindsight observation that by combining features of those references, one may achieve the benefits achieved from the invention as described and claimed in the application. Such a hindsight observation is not consistent with the Federal Circuit’s requirement for “particular factual findings.”

CLAIM 16

Claim 16 features:

A method of controlling concurrent users of a distributed resource on a network, the resource limited to a maximum number of concurrent users, the method comprising the computer-implemented steps of:
receiving a request for the distributed resource from a client process for a user having a user identification;
determining a home location associated with the user identification, the home location indicating a unique host among a plurality of hosts that execute a corresponding plurality of local lock manager processes of a distributed lock manager process,
wherein each of the plurality of local lock manager processes may grant a lock on the same resource;
sending a request for a lock object for the distributed resource to a first local lock manager process of the distributed lock manager process, the request including the home location;
receiving the lock object for the distributed resource from a second local lock manager process executing on the unique host, if a number of outstanding locks granted by the second local lock manager process is less than a value of a local resource maximum defined for the second local lock manager process; and
providing access to the resource to the first client only in response to receiving the lock object. (emphasis added).

As explained above, *Troxel*, *Soltis*, and *Wikstrom* have not been properly combined; as a result, the rejection of Claim 16 under 35 U.S.C. § 103(a) cannot be properly maintained based on the improper combination of *Troxel*, *Soltis*, and *Wikstrom*. However, even if *Troxel*, *Soltis*, and *Wikstrom* were to be properly combined, at least the above-bolded elements of Claim 16 would still not disclosed, taught, or suggested by *Troxel*, *Soltis*, and *Wikstrom*, either individually or in combination.

As explained above with respect to Claim 16, *Troxel*, *Soltis*, and *Wikstrom* each fail to teach the concept of a local lock manager process. Consequently, *Troxel*, *Soltis*, and *Wikstrom*, either individually or in combination, fail to show numerous features of Claim 16 featuring a local lock manager process, such as:

“determining a home location associated with the user identification, the home location indicating a unique host among a plurality of hosts that execute a corresponding plurality of local lock manager processes of a distributed lock manager process,
wherein each of the plurality of local lock manager processes may grant a lock on the same resource;
sending a request for a lock object for the distributed resource to a first local lock manager process of the distributed lock manager process, the request including the home location;
receiving the lock object for the distributed resource from a second local lock manager process executing on the unique host, if a number of outstanding locks granted by the second local lock manager process is less than a value of a local resource maximum defined for the second local lock manager process”

Additionally, *Troxel* fails to suggest a determining a home location associated with a user identification, where the home location indicates a unique host among a plurality of hosts that execute a corresponding plurality of local lock manager processes of a distributed lock manager process. **Indeed, even though *Troxel* is currently cited to show numerous elements involving a home location, the Office Action acknowledges that *Troxel*, “fails to teach the home location indicating a unique host among a plurality of hosts that execute a**

corresponding plurality of local lock manager processes of a distributed lock manager process.” Therefore, it is submitted that since *Troxel* is admitted to lack any teaching of a home location as claimed, it logically follows that *Troxel* cannot teach or suggest any element involving a home location as claimed.

Perhaps in recognition of the failings of the teachings *Troxel* with respect to the requirements of the claim elements, a portion of *Soltis* cited to show the element of “determining a home location associated with the user identification, the home location indicating a unique host among a plurality of hosts that execute a corresponding plurality of local lock manager processes of a distributed lock manager process.” However, this portion merely discusses activities performed by clients 105A-N of FIG. 4 of *Soltis*, and does not cure the deficiencies of *Troxel* because there is no suggestion of (a) a home location associated with a user identification, (b) anything indicating a unique host amount a plurality of hosts, or (c) a plurality of local lock manager processes of a distributed lock manager process. Consequently, this element cannot be disclosed, taught, or suggested for this reason as well.

Further, Claim 16 recites, “wherein each of the plurality of local lock manager processes may grant a lock on the same resource.” As explained above with reference to Claim 8, neither *Troxel*, nor *Soltis*, nor *Wikstrom* disclose, teach, or suggest this element. Consequently, this element is not disclosed, taught, or suggested by any cited reference, either individually or in combination.

As at least one element of Claim 16 is not disclosed, taught, or suggested by *Troxel*, *Soltis*, or *Wikstrom*, either individually or in combination, it is respectfully submitted that Claim 16 is patentable over the cited art and is in condition for allowance.

CLAIM 20

As explained above, *Troxel*, *Soltis*, and *Wikstrom* have not been properly combined; as a result, the rejection of Claim 20 under 35 U.S.C. § 103(a) cannot be properly maintained based on the improper combination of *Troxel*, *Soltis*, and *Wikstrom*.

For the second time, the Office Action has rejected Claim 20 under the same rationale as Claim 16. However, **Independent Claim 20 features elements that are not recited in Claim 16.** For example, Claim 20 features the elements of:

A method of controlling concurrent users of a distributed resource on a network, the distributed resource limited to a maximum number of concurrent users, the method comprising the steps of:
receiving at a first local lock manager process of a distributed lock manager process a request for a lock object for the distributed resource from a resource server, wherein the request includes data indicating a particular user home location;
determining whether a second local lock manager process of the distributed lock manager process is associated with the particular user home location, and if so, requesting the lock object from the second local lock manager process
wherein the first local lock manager process may grant a lock on the same resource as the second local lock manager process. (emphasis added)

The above-bolded elements of Claim 20 are not recited in Claim 16. As a result, **there are no reasons on the record as to why the above-bolded elements of Claim 20 are not patentable over the cited art.**

The Office cannot properly make a rejection “final” when no arguments have been presented on the record as to why the above elements of Claim 20 are shown by the cited art. Consequently, if the patent application is not allowed in the next communication, the Applicants respectfully request that the finality of the Office Action of June 1, 2006 be removed.

Troxel, *Soltis*, and *Wikstrom*, either individually or in combination, fail to suggest a first local lock manager process that may grant a lock on the same resource as the second local lock manager. Further, *Troxel*, *Soltis*, and *Wikstrom*, either individually or in combination, fail to

suggest a request that includes data indicating a particular user home location. As a result, *Troxel*, *Soltis*, and *Wikstrom*, either individually or in combination, cannot show the above bolded elements.

As a result, for at least the above reasons, the Applicant respectfully submits that Claim 20 is patentable over the cited art and is in condition for allowance.

CLAIM 25

Claim 25 features:

A method of distributing a resource on a network, the resource limited to a maximum number of concurrent users, the method comprising the steps of:
providing a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts,
wherein each of the plurality of local lock manager processes may grant a lock on the same resource;
generating a value for a local resource maximum number of users stored on each host of the plurality of hosts such that a summation over the plurality of hosts of the value for the local resource maximum yields an aggregate value that does not exceed the maximum number of concurrent users;
determining whether to increase a first value in a first resource maximum stored on a first host of the plurality of hosts; and
if it is determined to increase the first resource maximum, then
decreasing by a particular amount a second value in a second resource maximum stored on a second host of the plurality of hosts, and
increasing by the particular amount the first value in the first resource maximum stored on the first host,
wherein each local lock manager process is configured to grant a lock for the resource if the number of outstanding locks granted by the local lock manager process is less than a value of the local resource maximum stored on the corresponding host.
(emphasis added).

As explained above, *Troxel*, *Soltis*, and *Wikstrom* have not been properly combined; as a result, the rejection of Claim 25 under 35 U.S.C. § 103(a) cannot be properly maintained based on the improper combination of *Troxel*, *Soltis*, and *Wikstrom*. However, even if *Troxel*, *Soltis*,

and *Wikstrom* were to be properly combined, at least the above-bolded elements of Claim 25 would still not disclosed, taught, or suggested by *Troxel*, *Soltis*, and *Wikstrom*, either individually or in combination.

Neither *Troxel* nor *Soltis* nor *Wikstrom* suggest a local lock manager process as claimed. Consequently, for at least the above reasons, neither *Troxel* nor *Soltis* nor *Wikstrom* show, *inter alia*, the elements of:

“providing a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts,
wherein each of the plurality of local lock manager processes may grant a lock on the same resource,
wherein each local lock manager process is configured to grant a lock for the resource if the number of outstanding locks granted by the local lock manager process is less than a value of the local resource maximum stored on the corresponding host”

Troxel is cited to show the element of “generating a value for a local resource maximum number of users stored on each host of the plurality of hosts such that a summation over the plurality of hosts of the value for the local resource maximum yields an aggregate value that does not exceed the maximum number of concurrent users.” However, the portion of *Troxel* cited (Abstract; FIG. 4D; Col. 1, lines 15-36) lack any suggestion any value, for a local resource maximum number of users, that is stored on each host of a plurality of hosts such that a summation over the plurality of hosts of the value yields an aggregate value that does not exceed the maximum number of concurrent users. Instead, the portion of *Troxel* cited shows a single, centralized server that provides a locking mechanism. Thus, *arguendo*, while such a single, centralized server might store a value for a maximum number of users that may access a particular resource (although the cited portion of *Troxel* does not states this), such a single, centralized server cannot store “a value for a local resource maximum number of users stored on each host of the plurality of hosts such that a summation over the plurality of hosts of the value

for the local resource maximum yields an aggregate value that does not exceed the maximum number of concurrent users” since a single, centralized server is not analogous to a plurality of hosts. Consequently, this element is not disclosed, taught, or suggested by *Troxel*.

Further, Claim 25 recites, “wherein each of the plurality of local lock manager processes may grant a lock on the same resource.” As explained above with reference to Claim 8, neither *Troxel*, nor *Soltis*, nor *Wikstrom* disclose, teach, or suggest this element. Consequently, this element is not disclosed, taught, or suggested by any cited reference, either individually or in combination.

As at least one element of Claim 25 is not disclosed, taught, or suggested by *Troxel* or *Soltis* or *Wikstrom*, either individually or in combination, it is respectfully submitted that Claim 25 is patentable over the cited art and is in condition for allowance.

CLAIMS 9-15, 17-19, 21-24, 26-30, 32-35, 37-40, AND 42-44

Claims 32-35 are computer-readable medium claims that each feature limitations similar to those recited in method Claims 8, 16, 20, and 25. Consequently, it is respectfully submitted that Claims 32-35 are patentable over the cited art and are in condition for allowance for at least the reasons given above with respect to Claims 8, 16, 20, and 25 respectively.

Claims 37-40 are apparatus claims in accordance with 35 U.S.C. § 112, sixth paragraph, which each feature limitations similar to those recited in method Claims 8, 16, 20, and 25. Consequently, it is respectfully submitted that Claims 37-40 are patentable over the cited art and are in condition for allowance for at least the reasons given above with respect to Claims 8, 16, 20, and 25 respectively.

Claims 42-44 are apparatus claims that each feature limitations similar to those recited in method Claims 8, 16, and 25. Consequently, it is respectfully submitted that Claims 42-44 are patentable over the cited art and are in condition for allowance for at least the reasons given above with respect to Claims 8, 16, and 25 respectively.

Claims 9-15, 17-19, 21-24, and 26-30 are dependent claims, each of which depends (directly or indirectly) on one of the claims discussed above. Each of Claims 9-15, 17-19, 21-24, and 26-30 is therefore allowable for the reasons given above for the claim on which it depends. In addition, each of Claims 9-15, 17-19, 21-24, and 26-30 introduces one or more additional limitations that independently render it patentable. However, due to the fundamental differences already identified, to expedite the positive resolution of this case a separate discussion of those limitations is not included at this time, although the Applicant reserves the right to further point out the differences between the cited art and the novel features recited in the dependent claims.

CONCLUSION

It is respectfully submitted that all of the pending claims are in condition for allowance and the issuance of a notice of allowance is respectfully requested. If there are any additional charges, please charge them to Deposit Account No. 50-1302.

The Examiner is invited to contact the undersigned by telephone if the Examiner believes that such contact would be helpful in furthering the prosecution of this application.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP



Christopher J. Brokaw

Reg. No. 45,620

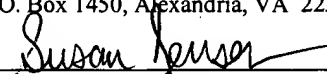
Date: October 6, 2006

2055 Gateway Place, Suite 550
San Jose, CA 95110
Telephone: (408) 414-1225
Facsimile: (408) 414-1076

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: **Mail Stop AF**, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450

on October 6, 2006 by


Susan Jensen